

DCA0214.1 - LABORATÓRIO DE ESTRUTURAS DE DADOS

Aulas 7 e 8: Listas sequenciais, encadeadas e suas
generalizações. Árvores e suas generalizações

Prof. Felipe Fernandes

26 e 27 Abril de 2019

— LEIA COM ATENÇÃO —

- I É proibido utilizar qualquer estrutura de dados ou algoritmos pré-fornecidos por alguma biblioteca C/C++.
 - II Em cada aula desta unidade, as listas de exercícios possuirão questões marcadas com \star . Todas as questões (de todas as aulas) marcadas com \star , devem ser submetidas via SIGAA até às 23h59 do 16 Maio 2019. Peso: 20% na nota de segunda unidade.
 - III O item II não lhe desobriga de resolver as questões sem \star . Todas as questões são importantes para a prova da segunda unidade.
-

- 1. Seja um polinômio da forma $P(x) = a_0x^n + a_1x^{n-1} + \dots + a_n$. Representar $P(x)$ através de uma lista encadeada conveniente e escrever algoritmos eficientes para efetuar as seguintes operações, onde $Q(x)$ é um outro polinômio.
 - (a) Dado um valor x , calcular $P(x)$
 - (b) $P(x) + Q(x)$
 - (c) $P(x) * Q(x)$
- 2. Seja A uma matriz esparsa $n \times m$, isto é, boa parte dos seus elementos são nulos ou irrelevantes. A expressão (1) fornece um exemplo de matriz esparsa. Faça o que se pede:

$$A = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 3 \\ 4 & 0 & 0 & 0 \end{bmatrix} \quad (1)$$

- (a) Implementar uma estrutura de dados que represente a matriz A e cujo espaço de armazenamento seja da ordem de $O(n + m + k)$, em vez de $O(nm)$, onde k é o total de elementos não nulos de A .
 - (b) Implementar um algoritmo eficiente para acessar o elemento $A[i][j]$ da matriz A representada pela estrutura do item (a).
 - (c) Dadas duas matrizes A e B , representadas pela estrutura do item (a), implementar um algoritmo eficiente que calcule a soma $A + B$.
3. ★ Implementar as operações de uma árvore binária de busca.
- (a) Busca
 - (b) Inserção
 - (c) Remoção
4. Uma árvore binária T , com n nós, é chamada de AVL se, para cada nó u de T , as alturas das subárvores esquerda e direita de u , respectivamente, $h_e(u)$ e $h_d(u)$, são tais que $|h_e(u) - h_d(u)| \leq 1$. Implemente um algoritmo com complexidade $O(n)$, que recebe uma árvore binária T , com n nós, e retorna verdadeiro se T é AVL ou falso caso contrário.
5. Implementar procedimentos **iterativos** para os percursos em pré-ordem, in ordem e pós-ordem de uma árvore binária.